The University of Georgia

ECSE 2920: Design Methodology

Deliverable 2

Group 11

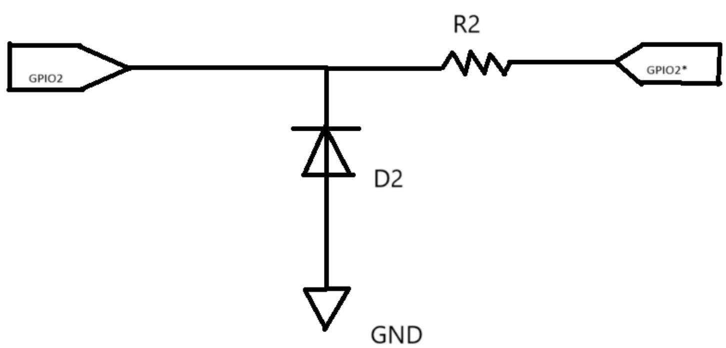
01/15/2025

Part 1: Diagram With GPIO2 on the PHP

Requirements: Our group is to draw a detailed diagram of GPIO2 on the PHP and explain how it works

The pi protection hat is used to protect the raspberry pi from electrical surges. Both components use GPIO pins. Specifically, the GPIO2 is a general-purpose input/output pin used on the circuit board. The GPIO2 is connected to pin 3 on the raspberry pi and the protection hat. This is represented by GPIO2(Raspberry pi GPIO stack - input) and GPIO2*(GPIO hat stack - output). Based off of the diagram, the 220-ohm resistor limits the amount of current traveling in or out of the GPIO pin to protect it from excessive current. The diode is connected to the ground to protect the pins from voltage spikes. The diode seems to be a Zener diode when comparing the pi protection hat diode to other diodes online. These diodes allow current to flow in both directions and specialize in current flowing in reverse when certain voltage is reached. If the voltage spikes, the diode will redirect the voltage to ground to avoid damaging the circuit. This enables the circuit to be protected from excessive current and voltage spikes.

Picture(s)



(GPIO2 Model)

Key Design Decision(s)

- What did your team clarify about the design?
    - We clarified the specific role of the GPIO2 pin as both an input and output signal within the raspberry pi and protection hat configuration.
- What were the competing choices in the design?
    - Not many competing choices in this design, our decision was based off the data sheets for the components
- Ultimately what did your team choose and why?
    - We chose to go with the pins on the GPIO that would not default to high so that the motors will not spin when no code is running. This will keep the same function while allowing us to minimize the risk of something breaking.

summary/part conclusion (make sure you address all parts of the requirements)

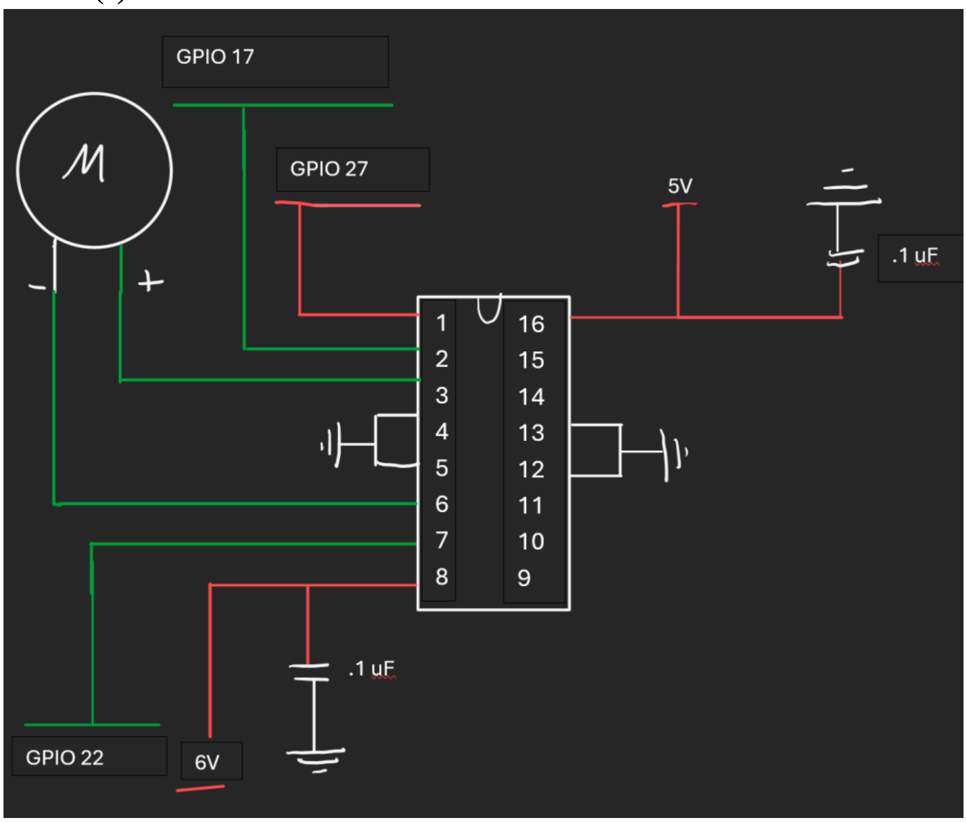
In conclusion we designed and made calculations based off of the hardware within the pi hat. The Zener diodes should protect it, but knowing which parts are in the circuit gives us more protection. We carefully looked over our pi hat to make sure we had the correct hardware written down for the next steps in our design.

<u>Part 2: H-Bridge circuit diagram</u>

Requirements: Create a detailed H-Bridge circuit diagram with RP4 and DC motor connections and components. Describe the function of the H-Bridge.

Pin 1 on the H-Bridge is the enable pin for pins 2 and 7. We are going to connect this to GPIO27 on the pi. When the output of this pin is set to high the output of the H-Bridge will match the input. When the GPIO pin is set to low the output of the H-Bridge will always be low. Pin 2 will connect to GPIO17 on the raspberry pi, when the enable is high and the GPIO17 is high the output at pin 3 of the H-Bridge, which is connected to the + side of the motor, will also be high. Pins 4, 5, 13, 12 are all ground and will also act as a heat sink. Pin 6 is connected to – side of the motor and will be controlled by pin 7 on the H-Bridge, which is connected to GPIO22. When pin 7 is high, pin 6 will also be high. Pin 8 is connected to a 6V source and will power the motor. Pin 16 is connected to 5V source to power the logic. Both pins 8 and 12 have bypass capacitors of .1uF as recommended by the data sheet. The remaining pins on the H-Bridge would be wired the same way to control the other motor. The pins on the pi that connect to the H-Bridge inputs would run the motor like such, when one pin is high and the other pin is low the motor will spin one way, and vice versa for the other way. When both pins are high and when both pins are low the motor will not spin.

Picture(s)



(H-Bridge Circuit Model)

Key Design Decision(s)

- What did your team clarify about the design?
  - The team clarified the exact GPIO pin configuration for controlling the H-bridge inputs and the enable pin
  - Clarified the use of bypass capacitors for power stabilization based on recommendations from the datasheet

    - Clarified the logic states to control the motor’s direction
- What were the competing choices in the design?
    - What to do with the enable pin. Should it permanently be high or should we connect it to a GPIO pin and be able to change the states.
    - Whether or not to add bypass capacitors to pins 8 and 16.
- Ultimately what did your team choose and why?
    - Deciding whether or not to add bypass capacitors to pins 8 and 16, led us to look at the data sheet which recommends that you do this, even though it would ultimately work without them. The team decided to add the bypass capacitors and air on the side of caution.
    - When deciding what to do with the enable pin, it doesn’t seem like a bad idea to have it always enabled but ultimately the group decided to connect it to a GPIO pin in case we need to be able to enable and disable the outputs.

TEST… Test… test

- What aspects of the design need to be tested?
    - Software?
  - *We have developed a testing program to assist us in getting our motor working with our H-Bridge.*
    - Hardware?
  - *Our upcoming testing will be done next week, and we will attempt to get our motor running with our H-Bridge.*
- Who is responsible for testing?
    - Tests ran?
    - Conclusions from testing?
  - *Testing will be conducted the following week.*

In conclusion we made the circuit diagram for the pi to H bridge to motor. We designed it in such a way that the motor can spin forward and backward and stop based on the outputs from the GPIO pins.

Part 3: Raspberry Pi setup

Requirements: Conduct the Raspberry Pi setup, and acquaint all members with working on pi/Github/etc. Include a photo reading “Hello World, from Group 11”.

During our meeting,

- *64-bit Raspberry Pi OS was installed and booted onto the RPi-4.*

- *Once that had been accomplished, we were able to successfully connect the pi to PAWS-Secure and ssh into it for remote access.*
- *Python, pip3, and any environment dependencies were installed.*
- *The "car" python environment was created to house and run our programs.*
- *Our GitHub repository was cloned onto the raspberry pi.*
- *The pi's bash script was modified to go to this environment, access our code directory, and print out the current directory tree on startup. This is for the convenience and wellbeing of every member on the team.*

Picture(s)

```
#Automatically goes to python venv and Group-11 Repository
workon car
echo "Hello, World! Here is the current code directory:"
cd Group-11/Code
tree

```

```
15/01/2025  14:19.39  /home/mobaxterm  ssh group11@172.20.190.110
Linux group11 6.6.51+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.51-1+rpt3 (2024-10-08) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jan 15 14:10:28 2025 from 172.20.182.203
spawn git pull
Username for 'https://github.com': mcs97515
Password for 'https://mcs97515@github.com':
Already up to date.

Hello World, from Group 11
Here is the current code directory:
.
├── Implementation
│   └── src
│       └── PseudoMovements.py
└── testScripts
    └── HelloWorldScripts
        ├── Cade_Toney.py
        ├── Dawson_Gulasa.py
        ├── Finn_Morton.py
        ├── Michael_See.py
        └── Nub_Ahmed.py

5 directories, 6 files
(car) group11@group11:~/Group-11/Code $
```

(Screenshot showing that we were able to ssh remotely to our pi & Hello World Statement)

Key Design Decision(s)

- What did your team clarify about the design?
  - *There needs to be common software file structure and practices.*
  - *Certain things need to happen at startup (Git Pull, Automatic file nav, etc.)*
- What were the competing choices in the design?
  - *Separate "Develop", "Test", and "Demo" directories*
  - *Singular Code Directory*

    - *Git Pull of Group 11 Repository on boot*
    - *Git Pull of entirety of Group 11*
- Ultimately what did your team choose and why?
    - *We chose to use have a degree of separation depending on code context. While that requires a better understanding of unix file navigation, it provides a protective boundary against accidentally running programs that may break the design.*
    - *We also decided to do a booting git pull of just our code directory in the group 11 repository. This is primarily for storage space concerns.*

TEST… Test… test

- What aspects of the design need to be tested?
    - Software? *Everyone needed to test remotely accessing the pi and to run the demo python files. We also needed to test the bash script modifications.*
    - Hardware? *Not much on hardware currently. Soon we plan to test GPIO pins using LED's.*
- Who is responsible for testing? *Michael See & Nub Ahmed*
    - Tests ran? *Everyone was able to ssh into the pi. The startup script modifications work well, and we can run python programs created from our laptops and uploaded/pulled from GitHub.*
    - Conclusions from testing? *The pi is setup. We are able to use the pi to test other hardware now.*

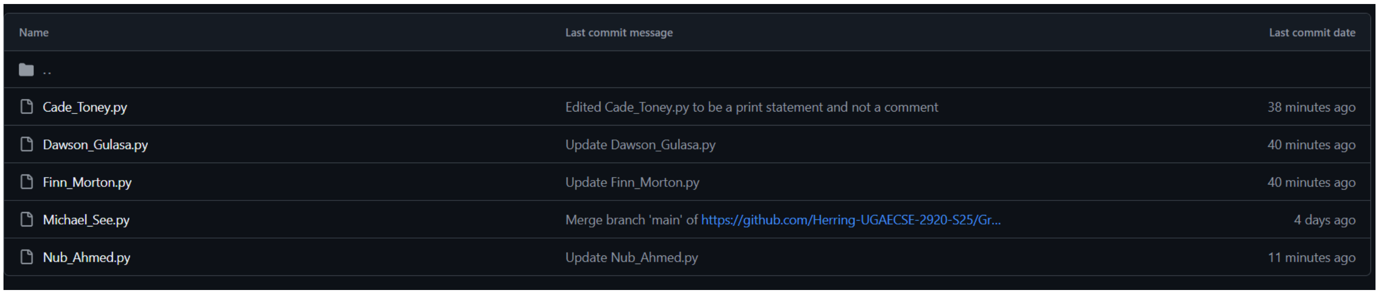
With the raspberry pi ready for programming/hardware interaction, it now serves as a tool to test the other components of this project. GPIO and motor testing are the current goals of next week, and once those tests are completed, we should be able to quickly write up specific sections of part A, as our goal is to get that out of the way as swiftly as possible.

Part 4: Commiting Changes

Requirements: Practice committing changes, making sure they are showing up within the repository. Each team member must push a test file to show you can do this.

Each team member has practiced committing changes on Github. We each committed a python file that contains a print statement of our names.

Picture(s)



Name	Last commit message	Last commit date
..		
Cade_Toney.py	Edited Cade_Toney.py to be a print statement and not a comment	38 minutes ago
Dawson_Gulasa.py	Update Dawson_Gulasa.py	40 minutes ago
Finn_Morton.py	Update Finn_Morton.py	40 minutes ago
Michael_See.py	Merge branch 'main' of https://github.com/Herring-UGAECSE-2920-S25/Gr...	4 days ago
Nub_Ahmed.py	Update Nub_Ahmed.py	11 minutes ago

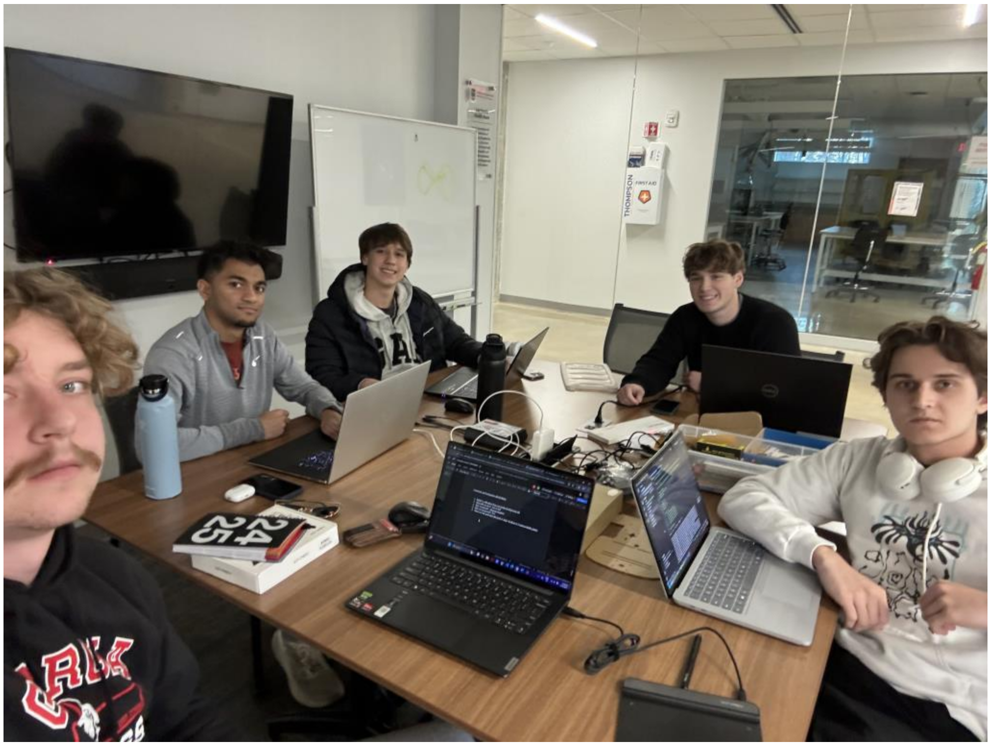
(Screenshot showing each member pushed a file on github)

summary/part conclusion (make sure you address all parts of the requirements)

- Everyone in our group committed at least one file to the Github. Specifically, a python file containing a print statement of our names.

Conclusion and Participation (REQUIRED)

1. **Include a selfie/photo from your group meeting/zoom call.**



2. When did you meet? 1/12 at 2:30
3. Who was present? All group members
4. Who was not present? N/A
5. What were the main ideas discussed or major decisions (1-2 sentences/bullet points)
    a. GitHub file structure and practicality of models were the main things discussed.
    b. Members were instructed on Raspberry Pi connection procedure